

# WEB SERVICE

**Alex Manzo**

*CILEA, Roma*

## *Abstract*

Negli ultimi anni, i maggiori produttori di software hanno focalizzato la loro attenzione sullo sviluppo e la standardizzazione di soluzioni mirate a integrare applicazioni web, indipendentemente dalla loro implementazione tecnologica. I conseguenti straordinari investimenti sono giustificati dalle nuove grandi opportunità offerta dai mercati B2B e B2C. I Web Service sono il risultato di tale sforzo e sono da considerare come il prossimo passo nell'evoluzione del World Wide Web.

During the last years most of the major software producers have focused on the development and standardization of solution aimed to integrate web applications, regardless of their implementation technology. The resulting remarkable investments are justified by the great new opportunities opened in the B2B and B2C markets. Web Service are the result of such efforts and are considered as the next step in the evolution of the World Wide Web.

*Keywords:* Webservice, WSDL, UDDI, Anagrafe Nazionale delle Ricerche.

## **Introduzione**

Secondo una definizione comune, i Web Service sono componenti software che interagiscono con altre applicazioni e sono accessibili in rete attraverso lo standard di comunicazione eXtensible Markup Language (XML) [1]. Essi non dipendono da alcun sistema operativo o linguaggio di programmazione e consentono ad altri servizi o applicazioni di raggiungerli e invocarli.

Tale meccanismo è basato su tre componenti, ognuno con un ben determinato ruolo: un fornitore di servizi (*Service Provider*) che implementa il servizio stesso e lo rende disponibile su Internet, un consumatore di servizi (*Service Requestor*) che utilizza un servizio esistente, inviando una richiesta XML e ricevendo una risposta XML, e un registro di servizi (*Service Register*) che fornisce uno spazio centralizzato dove gli sviluppatori possono pubblicare nuovi servizi o trovare quelli esistenti. Questi componenti si poggiano su determinate tecnologie quali XML, SOAP, WSDL e UDDI.

## **Architettura Web Service**

La necessità di far interoperare in modo dinamico e rapido i sistemi distribuiti ha portato oggi allo sviluppo di una nuova tecnologia: i Web

Service, che ampliano il concetto dei componenti dinamici tradizionali, quali applicazioni Cgi, Servlet Java, Script, ecc.

Estendendo la loro capacità di ottenere l'indipendenza dell'applicazione dal protocollo di trasporto e dalla piattaforma di implementazione, essi permettono l'integrazione di tutte le applicazioni esistenti su web, senza dover procedere a un *restyling* di parti di codice. Da quanto detto finora, possiamo definire un Web Service come un servizio disponibile su Internet che utilizza un sistema di scambio di messaggi standardizzato tramite XML e non è legato ad alcun sistema operativo o linguaggio di programmazione.

I Web Service rappresentano quindi un nuovo tipo di applicazioni indipendenti dalla piattaforma e dall'implementazione e aventi come caratteristiche primarie le capacità di poter essere:

- composti con altri servizi. Il servizio può a sua volta essere un client;
- esposti usando un linguaggio di descrizione del servizio. Un Web Service deve descrivere se stesso: quali tipi di richieste possono essere fatte, quali sono gli argomenti, quale è il trasporto;
- pubblicati in un registro di servizi. Un Web Service deve indicare a un registro di servizi

dove è localizzato (una sorta di “pagine gialle”);

- localizzati mediante un meccanismo standard (a *runtime* o a tempo di progetto). Un potenziale cliente deve trovare il Web Service in un registro dei servizi;
- invocati solitamente tramite la rete.

In generale, quindi, l'architettura di un Web Service si basa sull'interazione di tre componenti principali, come mostrato in figura 1:



Fig. 1 – Architettura Web Service

- il Service Provider, responsabile di compilare la descrizione del servizio (per esempio tutte le informazioni richieste per interagire con esso), di pubblicarla in uno o più Service Register, e di ricevere messaggi di invocazione del Web Service dal Service Requestor;
- il Service Requestor, responsabile di trovare la descrizione del servizio nel Service Register e di utilizzare i Web Service attraverso l'apertura di una connessione di rete, inviando una richiesta XML e ricevendo una risposta XML;
- i Service Register, il cui principale compito è quello di pubblicare le descrizioni dei Web Service per renderli disponibili al Service Requestor.

Scendendo nei particolari, una descrizione del servizio è formata principalmente da due parti distinte:

- *abstract interface*: descrive la logica del servizio in termini di messaggi che devono essere scambiati tra un Service Requestor e un Service Provider;
- *implementation interface*: contiene tutti i dettagli implementativi necessari per contattare e invocare i Web Service, come uno o più punti d'accesso (solitamente URL) alle differenti implementazioni dell'interfaccia astratta.

Per garantire l'interoperabilità delle operazioni di invocazione, scoperta e pubblicazione, è necessario un Web Service stack (figura 2), che fa uso di standard per ogni aspetto concettuale della tecnologia dei Web Service.

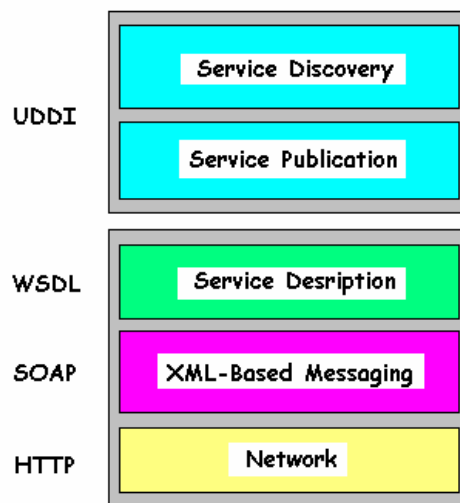


Fig. 2 – Web Service stack

## Tecnologie Web Service

Le tecnologie fondamentali per poter realizzare un Web Service sono quindi XML, SOAP, WSDL e UDDI.

### 1. XML

XML (eXtensible Markup Language) è un linguaggio che permette la rappresentazione dei dati in maniera indipendente dal dispositivo utilizzato (telefoni, palmari, ecc.). Esso consente di identificare attraverso un marcatore (*tag-element*) il tipo di dato, associando a quest'ultimo la relativa documentazione. Grazie alla separazione tra contenuto e presentazione dei dati, è la tecnologia ideale per spostare dati strutturati sulla rete.

I Web Service comunicano le loro informazioni in XML, il che comporta una serie di vantaggi.

Per prima cosa, XML ha il vantaggio di essere nato per consentire la comunicazione tra diverse piattaforme, quindi lo standard XML risolve già, a livello di informazione, tutte quelle problematiche di integrazione con sistemi eterogenei. Inoltre, XML ha la particolarità di essere un protocollo testuale, quindi facilmente leggibile dagli esseri umani.

Questo è un vantaggio in fase di debug dell'applicazione, in quanto il valore dei singoli campi è immediatamente visibile, ma ha lo

svantaggio di richiedere più potenza di elaborazione rispetto ai protocolli binari, più vicini alla macchina.

## 2. SOAP

SOAP (Simple Object Access Protocol) è un protocollo per lo scambio di messaggi XML tra le applicazioni. È un progetto open source in fase di standardizzazione presso il World Wide Web Consortium ([www.w3c.org](http://www.w3c.org)). SOAP definisce solo la struttura del messaggio e alcune regole per elaborarlo, rimanendo quindi a un alto livello e completamente indipendente dal protocollo di trasporto sottostante. Per il trasporto dei messaggi di richiesta e risposta utilizza il protocollo http, sul quale i parametri e i comandi sono codificati con il linguaggio XML.

SOAP è composto da tre parti:

- SOAP *envelope construct*: definisce la struttura del messaggio;
- SOAP *encoding*: definisce il meccanismo usato per lo scambio dei dati;
- SOAP *Rpc*: consente di invocare un metodo su un oggetto remoto (Remote Procedure Call).

Il principale obiettivo di SOAP è garantire semplicità ed estensibilità, sfruttando le caratteristiche dei sistemi di rete senza tener conto del protocollo. Un messaggio SOAP, che è un documento XML, deve contenere le seguenti parti, come mostrato in figura 3:

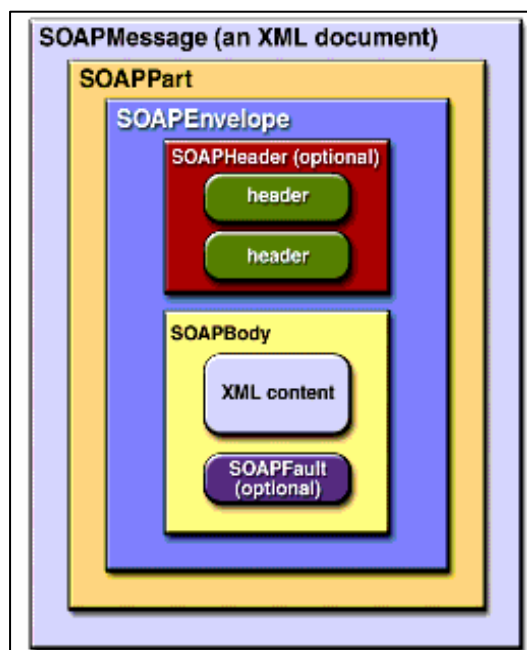


Fig. 3 – Struttura di un messaggio SOAP

- *Header* (opzionale): consente l'aggiunta di caratteristiche al messaggio SOAP, quali l'action, ecc.;
- *Envelope* (obbligatorio): costituisce l'elemento principale e rappresenta il messaggio all'interno del quale si trova il *Body* (obbligatorio). L'Envelope contiene quindi le informazioni necessarie al destinatario del messaggio e le informazioni sulla struttura e i contenuti dell'oggetto remoto richiesto.

## 3. WSDL

Il WSDL (Web Service Description Language) consente ai Service Provider di descrivere il formato delle richieste da inoltrare ai Web Service. Rappresenta il contratto tra cliente e fornitore di servizi.

Il file WSDL associato a un servizio descrive cosa esso può fare (metodi e parametri da invocare, valori di ritorno, ecc.), dove esso risiede e come invocarlo.

Inoltre, WSDL non è uno standard legato a un livello di trasporto particolare (come SOAP), ma è aperto all'utilizzo con protocolli differenti.

L'utilizzo di WSDL ha due vantaggi:

- alcuni strumenti di sviluppo sono in grado di generare, da un file WSDL, il codice che implementa l'infrastruttura per realizzare il servizio;
- con WSDL è possibile disaccoppiare il Web Service dal protocollo di trasporto e dai percorsi fisici.

Un documento WSDL è un file XML contenente un insieme di definizioni. L'aspetto principale è l'insieme di regole che consente di definire in modo astratto un servizio Web; oltre a ciò, sono presenti le specifiche per collegare il servizio a SOAP, HTTP o MIME.

Gli elementi che compongono una definizione WSDL sono:

- il tipo (*type*): contiene le definizioni dei tipi di dati non predefiniti;
- il messaggio (*message*): contiene i parametri di richiesta e di risposta del servizio;
- l'operazione (*portType*): descrive l'insieme delle operazioni supportate, ognuna delle quali costituita dai messaggi di richiesta e di risposta;
- i collegamenti (*binding*): descrivono il protocollo supportato, le operazioni consentite e i relativi input e output;
- la definizione del servizio (*Service*): rappresenta la locazione del binding a esso associato.

#### 4. UDDI

L'UDDI (Universal Description Discovery and Integration) permette di localizzare e conoscere i servizi offerti dalle aziende iscritte, come se fossero delle "pagine gialle" residenti sul web. UDDI si basa sul protocollo SOAP, in quanto sia le richieste che le risposte sono oggetti UDDI incapsulati all'interno di messaggi SOAP. Più precisamente, si tratta di una specifica per registri di informazione sui servizi web (quale per esempio i dettagli per collegarsi a esso): i fornitori di servizi e i servizi stessi sono descritti in XML.

I registri UDDI sono supportati da una rete mondiale di nodi, collegati tra di loro in una sorta di federazione, in modo simile alla tecno-

logia DNS. Quando un client sottopone un'informazione al registro, questo la propaga agli altri nodi. Si ottiene così la ridondanza dei dati, fornendo una certa affidabilità. Il ruolo del singolo nodo rimane fondamentale poiché, nel momento in cui un client gli sottopone dei dati, questo ne diviene il proprietario e sarà in futuro il solo a poter operare importanti operazioni sui dati, quali la loro eliminazione.

Lo scenario nel quale trova pieno utilizzo l'UDDI può essere ricondotto a tre fasi fondamentali: pubblicazione, ricerca, collegamento. Le informazioni gestite dal registro UDDI sono di tre tipi, come illustrato in figura 4.

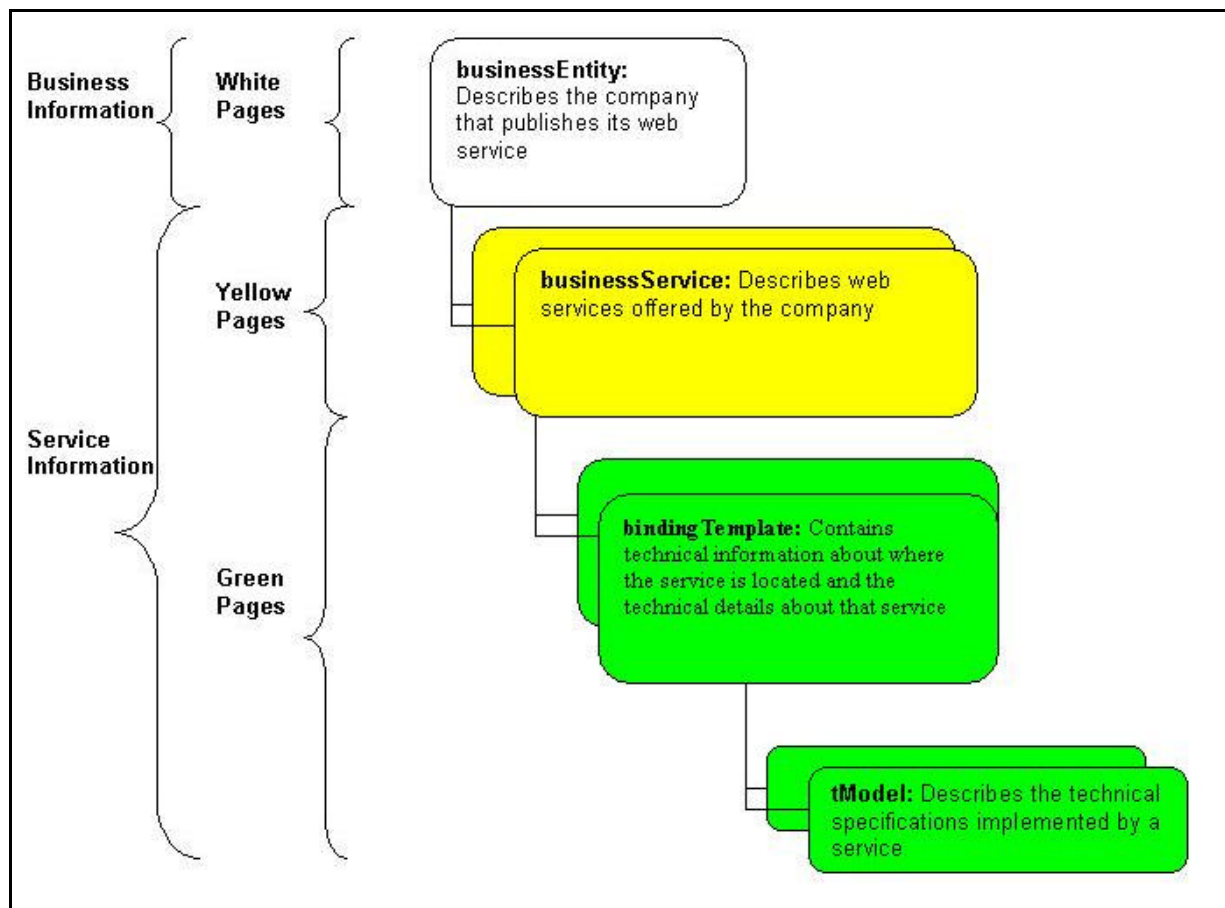


Fig. 4 – Le informazioni contenute nei registri UDDI

Più in dettaglio:

- pagine bianche: contengono informazioni anagrafiche delle aziende, come l'indirizzo e i numeri di telefono;
- pagine gialle: definiscono la categorizzazione dei servizi e delle aziende. Un'azienda potrebbe essere infatti catalogata in base alla sua posizione geografica e al proprio settore industriale;
- pagine verdi: costituiscono le informazioni tecniche dei servizi, quelle utili a livello informatico. Queste riguardano URL dei servizi o gli eventuali documenti WSDL.

### I Web Service nell'Anagrafe Nazionale delle Ricerche

Visti i vantaggi della tecnologia sopra descritta, i Web Service sono uno dei componenti tecnologici usati nel sistema informatico dell'Anagrafe Nazionale delle Ricerche [4]. Infatti, con questo mezzo si sono esposti in modo indipendente dalla piattaforma, e quindi accessibile da varie applicazioni, i dati dei progetti presenti in alcuni database del CILEA e del CINECA tutti inglobati nel metamotore di ricerca Me.Mo.R.I. Si sono così implementati Web Service per ogni database con diverse tecnologie (java e .net), richiamati attraverso un unico punto d'accesso, il metamotore. Tale metamotore usa un protocollo di trasmissione, definito dal CILEA per comunicare con le interfacce di interrogazione dei singoli database, esposti come Web Service.

In Me.Mo.R.I., inoltre, la trasmissione dei dati è veicolata tramite canali crittografati, in modo da garantire la sicurezza dei dati stessi.

Il Metamotore quindi fa uso di un Certificato, rilasciato da un Autorità di Certificazione, che garantisce allo stesso di essere riconosciuto come tale dai Web Service, che a loro volta espongono un certificato per essere identificati dal Metamotore. Solo dopo il riconoscimento, il metamotore è in grado di fare uso dei servizi offerti dai vari Web Service.

### Bibliografia

- [1] S. Graham, S. Simeonov, T. Boubez, D. Davis, G. Daniels, Y. Nakamura, R. Neyama SAMS, *Building Web Service with Java: Making Sense of XML, SOAP, WSDL and UDDI*.
- [2] Web Service – SOAP  
URL: <http://ws.apache.org/soap>
- [3] Web Service Conceptual Architecture  
URL: <http://www.3.ibm.com/software/solutions/webService/pdf/WSCA.pdf>
- [4] D. De Vita, *MEMORI: Il MetaMotore della Ricerca Italiana*, Bollettino del CILEA, n. 104, dicembre 2006.